

Exam 1 – Spring 2012 – Solution Notes

1. a. The image concerns Myers' recommendation that meeting participants agree in advance to a prohibition on personal attacks and "put-downs."
b. The image is used to illustrate the clear advantages in having two shorter meetings instead of one longer meeting when two separate topics are of interest to overlapping subsets of the attendees. Myers' specific recommendation is to keep meetings to one topic in order to keep them small, short, and relevant to all in attendance.
2. e
3. a. By "missing", he means a non-existent path *that is required or necessary for the program to meet its requirements*.
b. Suppose the statement $X := Y/Z$ lies along a path and that it is possible for Z to equal 0. This "data sensitivity error" might not be detected unless the path happened to be executed with inputs that result in Z equaling 0 when the statement is encountered. Thus, exhaustive path testing may not detect such errors. See page 8 of the Myers reading for another example.
4. The PURPOSE of regression testing is to detect unexpected problems with program functionality that worked correctly in a previous version caused by subsequent program changes.
5. f
6. Consider the program: If (A) then s1
If (B OR C) then s2

and the following 4 test case primitive condition values:

	A	B	C	path
test 1	T	T	T	TT
test 2	F	F	F	FF
test 3	T	T	F	TT
test 4	T	F	F	TF

Test cases 1 and 2 alone obviously provide Condition Coverage, but not Basis Paths coverage since $C=3$ distinct program paths are required. Therefore, Condition Coverage does not subsume Basis Paths Coverage. (cont'd.)

Test cases 2, 3, and 4 alone provide Basis Paths Coverage ($C=3$ distinct, simple paths that cover all branches), but not Condition Coverage since condition C is never true. Therefore, Basis Paths Coverage does not subsume Condition Coverage.

Therefore, Condition Coverage and Basis Paths Coverage are independent.

7. (1) Investigate what it would take to implement an effective inspections program.
 (2) Launch an effort to determine what (testing) tools would provide the most leverage.
 (3) Begin today to cultivate a "test to break" attitude of creative destruction.
8. a. Per Fagan, "The Hawthorne Effect is a psychological phenomenon usually experienced in human-involved productivity studies. The effect is manifested by participants producing above normal because they know they are being studied."
 b. A *control sample* was selected at random from many pieces of work after inspections had become accepted as commonplace in the development environment and differences in productivity between the experimental sample and the control sample were compared (and found to be insignificant).
9. +HUGE_VAL, +TINY_VAL, 0, -TINY_VAL, -HUGE_VAL

10. b

11. The seemingly "hyperbolic step" is initially RUNNING (not just implementing) the test BEFORE the code is actually implemented:



12. Number of test cases required for Strong Equivalence Class Testing: 24
 Number of test cases required for Weak Equivalence Class Testing: 4
13. a. 4, 4, 4, 4 b. 9, 5, 3, 3 c. 8, 4, 2, 1 d. infinite, 3, 1, 1
14. c
15. In Usability Test, "protocol analysis" (as described in class) involves carefully monitoring (usually video-recording) experimental subjects who are using systems in a realistic work environment. Test subjects are often asked to verbalize their thoughts/reactions during the work session. Careful analysis after the fact by HCI experts often reveal interface design problems or shortcomings that impede productivity or work *flow* – i.e., "usability bottlenecks".
16. An "invalid equivalence class" is simply an equivalence class of invalid or unexpected input values that are (nominally) associated with some specified, "exceptional" program behavior (e.g., outputting an error message and halting). Such classes should be covered "one at a time" because the program's correct or incorrect handling of such inputs can *mask* the detection/handling of other detected or undetected invalid inputs encountered in the test.

17. a. One may need a **driver** to take the place of A, a **stub** to take the place of G, and an **oracle** to assist in determining if C and K execute together in accordance with requirements.
 b. Yes, scaffolding may be required whenever program elements are tested in the absence of other program elements that they call or are called by, or whenever another program (an oracle) is required to determine if the element(s) is/are executing in accordance with requirements.
18. (1) the inherently state-dependent behavior of methods -- testers must take both program "outputs" and changes in object state into account.
 (2) encapsulation of methods and state – testers may need to use special instrumentation when methods result in hidden state changes, or are themselves private.
 (3) inheritance – testers must decide which methods in a derived class need to be (re-)tested.
 (4) polymorphism and dynamic binding -- testers must be aware of the run time bindings that may occur, and determine which methods (or combinations of methods) to test.
19. e
20. a. ii
 b. i (Clearly AEMC \neq "Strategy #3 Plus Culling Rules". To see that the converse is also false, consider the C-E relationships $C1 \Rightarrow E1$ and $C2 \Rightarrow E1$, and the constraint $C1 \Leftrightarrow C2$. In this case, the only feasible combination that would result in coverage of E1 (i.e., $C1 \text{ AND } C2$) would be culled.)
21. (35) $\equiv \neg 1, \neg B$
 $\neg B \equiv 4, \neg A$
 $\neg A \equiv \frac{(2, 3)}{\text{infeas}} \vee \frac{(2, \neg 3)}{\text{OK}} \vee \frac{(\neg 2, 3)}{\text{OK}}$
 $\equiv \neg 1, 2, \neg 3, 4$
 $\neg 1, \neg 2, 3, 4$

CAUSES	TEST CASES									
	1	2	3	4	5	6	7	8	9	10
$2 \leq \text{no. chars} \leq 8$ (1)	F	F								
1 st char is letter (2)	T	F								
1 st char is \$ (3)	F	T								
others letters/digits (4)	T	T								
EFFECT										
output J12 only (34)	T	T								

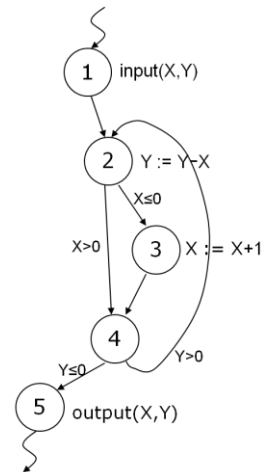
22. O, W, D, C, G, K, S, F, L, H, I

23. a.

```

1. input(X,Y)
   repeat
2.   Y := Y-X
   if X<=0 then
3.     X := X+1
4.   end_if_then
   until (Y<=0)
5. output(X,Y)

```



b.

<u>du-pair</u>	<u>du-path(s)</u>
(1,2)	<1,2>, <1,2,4,2>
(1,3)	<1,2,3>, <1,2,4,2,3>
(1,5)	<1,2,4,5>
(1,<2,3>)	<1,2,3>
(1,<2,4>)	<1,2,4>
(3,2)	<3,4,2>
(3,3)	<3,4,2,3>
(3,5)	<3,4,5>, <3,4,2,4,5>
(3,<2,3>)	<3,4,2,3>
(3,<2,4>)	<3,4,2,4>

c. $X \leq 0 \wedge Y - X > 0 \wedge X + 1 > 0 \wedge (Y - X) - (X + 1) \leq 0$ d. $X = 0$ and $Y = 1$

24. No, it was not an error. Edge <5,6>, which represents the same while loop "False Branch" as edge <1,6>, is covered -- as are all other program "Branches". (The definitions of some dataflow coverage criteria are based on a control-flow graph with this "feature.")

25. An operational profile describes how a system is expected to be used in a particular end-user environment.